User's Guide for the NMM Core of the Weather Research and Forecast (WRF) Modeling System Version 2.1

Chapter 5: WRF NMM Model

Table of Contents

Introduction

The WRF-NMM model is a fully compressible, non-hydrostatic model with a hydrostatic option (Janjic et al. 2001; Janjic 2003a, Janjic 2003b). The model uses a terrain following hybrid sigma-pressure vertical coordinate. The grid staggering is the Arakawa E-grid. The same time step is used for all terms. The dynamics conserve a number of first and second order quantities including energy and enstrophy (Janjic 1984).

The WRF-NMM model code contains an initialization program ***(real_nmm.exe***; see Chapter 4) and a numerical integration program (***wrf.exe***). The WRF-NMM model Version 2.1 supports a variety of capabilities. These include:

- Real-data simulations
- Non-hydrostatic and hydrostatic (runtime option)
- Applications ranging from meters to thousands of kilometers

WRF-NMM Dynamics in a Nutshell:

Time stepping:

Horizontally propagating fast-waves:	Forward-backward scheme
Vertically propagating sound waves:	Implicit scheme

Horizontal:	Adams-Bashforth scheme

Vertical:	Crank-Nicholson scheme
TKE, water species:	Explicit, iterative, flux-corrected (called every two time steps).

Advection (space) for T, U, V:

Horizontal: Energy and enstrophy conserving, quadratic conservative, second order
Vertical: Quadratic conservative, second order
TKE, Water species: Upstream, flux-corrected, positive definite, conservative

Diffusion

Diffusion in the WRF-NMM is categorized as lateral diffusion and vertical diffusion. The vertical diffusion in the PBL and in the free atmosphere is handled by the surface layer scheme and the Mellor-Yamada-Janjic scheme (Janjic 1996a, 1996b, 2002a, 2002b). The lateral diffusion is formulated following the Smagorinsky non-linear approach (Janjic 1990). The control parameter for the lateral diffusion is the square of Smagorinsky constant.

Divergence damping:

The horizontal component of divergence is damped (Sadourny 1975). In addition, if applied, the technique for coupling the elementary subgrids of the E grid (Janjic 1979) damps the divergent part of flow.

Physics Options

All available WRF System physics package options are listed below. Some of these options have not yet been tested for WRF-NMM. Indication of the options that have been tested, as well as the level of the testing, is included in the discussion below.

Microphysics *(mp_physics)*

Note: The Ferrier scheme is currently the only microphysics scheme that works with WRF-NMM. Changes will be made in future releases to accommodate the other microphysics options.

0. No microphysics

1. Kessler scheme: A warm-rain (i.e. no ice) scheme used commonly in idealized cloud modeling studies. (Kessler, 1969; Wicker and Wilhemson, 1995)

2. Lin et al. scheme: A sophisticated scheme that has ice, snow and graupel processes, suitable for real-data high-resolution simulations (Lin et al. 1983, Rutledge and Hobbs 1984, Tao et al. 1989, Chen and Sun 2002).

3. WRF Single-Moment (WSM) 3-class simple ice scheme: A simple efficient scheme with ice and snow processes suitable for mesoscale grid sizes (Hong et al. 1998, Hong et al. 2004).

4. WRF Single-Moment (WSM) 5-class scheme. A slightly more sophisticated version of option 3 that allows for mixed-phase processes and super-cooled water (Hong et al. 1998, Hong et al. 2004).

5. Ferrier scheme: A scheme that includes prognostic mixed-phase processes (Ferrier et al. 2002). This scheme was recently changed so that ice saturation is assumed at temperatures colder than -30_C rather than -10_C as in the original implementation. (This scheme is well tested for WRF-NMM, used operationally at NCEP.)

6. WSM 6-class graupel scheme: A new scheme with ice, snow and graupel processes suitable for high-resolution simulations (Lin et al. 1983, Dudhia 1989, Hong et al. 1998).

8. Thompson et al. scheme: A scheme with six classes of moisture species plus number concentration for ice as prognostic variables. (Thompson et al. 2004)

98. NCEP 3-class simple ice scheme (to be removed): An older version of WSM-3.

99. NCEP 5-class scheme (to be removed): An older version of WSM-5.

Longwave Radiation *(ra_lw_physics)*

1. RRTM scheme: Rapid Radiative Transfer Model. An accurate scheme using look-up tables for efficiency. Accounts for multiple bands, trace gases, and microphysics species (Mlawer et al. 1997). (Preliminary testing for WRF-NMM.)

99. GFDL scheme: Geophysical Fluid Dynamics Laboratory (GFDL) Longwave. An older version multi-band, transmission table look-up scheme with carbon dioxide, ozone and water vapor absorptions (Fels and Schwarzkopf 1975, Schwarzkopf and Fels 1985, Schwarzkopf and Fels 1991). Cloud microphysics effects are included. (This scheme is well tested for WRF-NMM, used operationally at NCEP.)

Shortwave Radiation *(ra_sw_physics)*

1. Dudhia scheme: Simple downward integration allowing for efficient cloud and clear-sky absorption and scattering (Dudhia 1989).

2. Goddard Shortwave scheme: Two-stream multi-band scheme with ozone from climatology and cloud effects (Chou and Suarez 1994).

99. GFDL scheme: Geophysical Fluid Dynamics Laboratory (GFDL) shortwave. A two spectral bands, k-distribution scheme with ozone and water vapor as the main absorbing gases (Lacis and Hansen 1974). Cloud microphysics effects are included. (This scheme is well-tested for WRF-NMM, used operationally at NCEP.)

Surface Layer (sf_sfclay_physics)

1. Monin-Obukhov Similarity scheme: Based on Monin-Obukhov with Carslon-Boland viscous sub-layer and standard similarity functions from look-up tables (Skamarock et al. 2005).

2. Janjic Similarity scheme: Based on similarity theory with viscous sublayers both over solid surfaces and water points (Janjic, 1996b, Chen et al. 1997). (This scheme is well tested for WRF-NMM, used operationally at NCEP.)

3. NCEP Global Forecasting System (GFS) scheme: The Monin-Obukhov similarity profile relationship is applied to obtain the surface stress and latent heat fluxes using a formulation based on Miyakoda and Sirutis (1986) modified for very stable and unstable situations. Land surface evaporation has three components (direct evaporation from the soil and canopy, and transpiration from vegetation) following the formulation of Pan and Mahrt (1987). (This scheme has been tested by NCEP for the WRF-NMM.)

Land Surface (sf_surface_physics)

1. Thermal Diffusion scheme: Soil temperature only scheme, using five layers (Skamarock et al. 2005).

2. NOAH Land-Surface Model: Unified NCEP/NCAR/AFWA scheme with soil temperature and moisture in four layers, fractional snow cover and frozen soil physics. (Chen and Dudhia, 2001)

3. RUC Land-Surface Model: Rapid Update Cycle operational scheme with soil temperature and moisture in six layers, multi-layer snow and frozen soil physics (Smirnova et al. 1997, 2000).

99. NMM Land Surface Scheme: The NMM LSM package is based in the pre-May 2005 NOAH Land Surface Model (LSM) in the operational NAM/Eta with soil temperature and moisture in 4 layers, fractional snow cover and frozen soil physics (Ek et al. 2003) and is quite similar to the unified NOAH LSM (option 2 above). (This scheme is well tested for WRF-NMM, used operationally at NCEP.)

Planetary Boundary Layer *(bl_pbl_physics)*

1. Yonsei University scheme (YSU): Next generation MRF-PBL. Non-local-K scheme with an explicit entrainment layer and parabolic K profile in unstable mixed layer (Skamarock et al. 2005). (Preliminary testing for WRF-NMM.)

2. Mellor-Yamada-Janjic Scheme: One-dimensional prognostic turbulent kinetic energy scheme with local vertical mixing (Janjic 1990, 1996a, 2002). (This scheme is well-tested for WRF-NMM, used operationally at NCEP.)

3. NCEP Global Forecast System scheme: First-order vertical diffusion scheme of Troen and Mahrt (1986) further described in Hong and Pan (1996). The PBL height is determined using an iterative bulk-Richardson approach working from the ground upward whereupon the profile of the diffusivity coefficient is specified as a cubic function of the PBL height. Coefficient values are obtained by matching the surface-layer fluxes. A counter-gradient flux parameterization is included. (This scheme has been tested by NCEP for WRF-NMM.)

99. MRF scheme: An older version of YSU with implicit treatment of entrainment layer as part of non-local-K mixed layer (Hong and Pan 1996).

Cumulus Parameterization *(cu_physics)*

0. No cumulus parameterization. (Tested for WRF-NMM)

1. Kain-Fritsch scheme: Deep and shallow sub-grid scheme using a mass flux approach with downdrafts and CAPE removal time scale (Kain 2004, Kain and Fritsch 1990, 1993). (Preliminary testing for the WRF-NMM.)

2. Betts-Miller-Janjic scheme: Adjustment scheme for deep and shallow convection relaxing towards variable temperature and humidity profiles determined from thermodynamic considerations (Janjic 1994, 2000). (This scheme is well tested for WRF-NMM, used operationally at NCEP.)

3. Grell-Devenyi ensemble scheme: Multi-closure, multi-parameter, ensemble method with typically 144 sub-grid members (Grell and Devenyi 2002).

4. Simplified Arakawa-Schubert scheme: Penetrative convection is simulated following Pan and Wu (1995), which is based on Arakawa and Schubert (1974) as simplified by Grell (1993) and with a saturated downdraft. (This scheme is well tested for WRF-NMM by NCEP.)

<u>Below is a summary of physics options that are well-tested for WRF-NMM:</u>

&physics	Identifying Number	Physics options
mp_physics (max_dom)	5	Microphysics-Ferrier
ra_lw_physics	99	Long-wave radiation - GFDL (Fels-Schwarzkopf)
ra_sw_physics	99	Short-wave radiation - GFDL (Lacis-Hansen)

sf_sfclay_physics	2	Surface-layer: Janjic scheme
sf_surface_physics	99	Land-surface – NMM LSM
bl_pbl_physics	2	Boundary-layer - Mellor-Yamada-Janjic TKE
cu_physics	2	Cumulus - Betts-Miller-Janjic scheme
num_soil_layers	4	Number of soil layers in land surface model-4.

Description of Namelist Variables

The settings in the "***namelist.input***" file are used to configure WRF-NMM. This file should be edited to specify: dates, time step, domain size, output options, and physics options. When modifying the "***namelist.input***" file, be sure to take into account the following points:

"***time_step":*** There is no simple formula for determining the time step for **WRF-NMM**. The following are pre-tested time-steps:

Approximate Grid Spacing (km)	DELTA_X (in degrees)	DELTA_Y (in degrees)	Time Step (seconds)
4	0.026726057	0.026315789	9-10s
8	0.053452115	0.052631578	18s
10	0.066666666	0.065789474	24s
12	0.087603306	0.075046904	25-30s
22	0.154069767	0.140845070	60s
32	0.222222222	0.205128205	90s

"e_we and e_sn": Given WRF-NMM's E-grid staggering, the end index in the east-west direction (***e_we***) and the south-north direction (***e_sn***) need to be set with care. The simple rule is:

e_we = ***XDIM***+1,
e_sn = ***YDIM***+1,

Example:

If ***XDIM*** and ***YDIM*** are set up as follows in your ***wrfsi.nl***

XDIM = 123,
YDIM = 201,

then ***e_we*** and ***e_sn*** in ***namelist.input*** should be:

e_we = 124,
e_sn = 202.

"dx and dy": For WRF-NMM, ***dx*** and ***dy*** are the horizontal grid spacing in degrees, rather than meters (unit used for WRF-ARW). Note that ***dx*** should be slightly larger than ***dy*** due to the convergence of meridians approaching the poles on the rotated grid.

If ***MOAD_DELTA_X*** and ***MOAD_DELTA_Y*** are set up as follows in your ***wrfsi.nl:***

MOAD_DELTA_X = .0534521,
MOAD_DELTA_Y = .0526316,

then ***dx*** and ***dy*** in ***namelist.input*** should be:

dx = .0534521,
dy = .0526316.

As can be seen from the above example:

dx = MOAD_DELTA_X
dy = MOAD_DELTA_Y

e_we = XDIM+1
e_sn = YDIM+1

For more information about the horizontal grid spacing for WRF-NMM, please see Chapter 3, WRF-NMM SI

"nio_tasks_per_group": The number of ***I/O*** tasks ***(nio_tasks_per_group)*** should evenly divide into the number of compute tasks in the ***J-direction*** on the grid (that is the value of ***nproc_y***). For example, if there are 6 compute tasks in the ***J-direction***, then ***"nio_tasks_per_group"*** could legitimately be set to 1, 2, 3, or 6. The user needs to use a number large enough that the quilting for a given output time is finished before the next output time is reached. If one had 6 compute tasks in the ***J-direction*** (and the number in the ***I-direction*** was similar), then one would probably choose either 1 or 2 quilt tasks.

The following table provides an overview of the parameters specified in ***namelist.input***. Note that "***namelist.input***" is common for both WRF cores (WRF-ARW and WRF-NMM). Most of the parameters are valid for both cores. However, some parameters are only valid for one of the cores. Core specific parameters are noted in the table. In addition, some physics options have not been tested for WRF-

NMM. Those options that have been tested are highlighted by indicating whether they have been "fully" or "preliminarily" tested for WRF-NMM.

Variable Names	**Value (Example)**	**Description**
&time_control		Time control
run_days	2	Run time in days
run_hours	0	Run time in hours Note: If run time is more than 1 day, one may use both run_days and run_hours or just run_hours. e.g. if the total run length is 36 hrs, you may set run_days = 1, and run_hours = 12, or run_days = 0, and run_hours 36.
run_minutes	00	Run time in minutes
run_seconds	00	Run time in seconds
start_year (max_dom)	2005	Four digit year of starting time
start_month (max_dom)	04	Two digit month of starting time
start_day (max_dom)	27	Two digit day of starting time
start_hour (max_dom)	00	Two digit hour of starting time
start_minute (max_dom)	00	Two digit minute of starting time
start_second (max_dom)	00	Two digit second of starting time
end_year (max_dom)	2005	Four digit year of ending time
end_month (max_dom)	04	Two digit month of ending time
end_day (max_dom)	29	Two digit day of ending time
end_hour (max_dom)	00	Two digit hour of ending time
end_minute (max_dom)	00	Two digit minute of ending time
end_second (max_dom)	00	Two digit second of ending time
interval_seconds	10800	Time interval between incoming real data, which will be the interval between the lateral boundary condition files. This parameter is only used by "*real_nmm.exe*".
history_interval (max_dom)	60	History output file interval in minutes
frames_per_outfile (max_dom)	1	Output times per history output file, used to split output files into smaller pieces
restart	.false.	Logical indicating whether run is a restart run

Variable Names	**Value (Example)**	**Description**
restart_interval	60	Restart output file interval in minutes
io_form_history	2	2 = netCDF
io_form_restart	2	2 = netCDF
io_form_input	2	2 = netCDF
io_form_boundary	2	1. Binary format 2. netCDF format 4. PHD5 format 5. GRIB-1 format
debug_level	0	0 - for standard runs, no debugging. 1 - netcdf error messages about missing fields. 50,100,200,300 values give increasing prints. Large values trace the job's progress through physics and time steps.
&Domains		Domain definition
time_step	18	Time step for integration in integer seconds
time_step_fract_num	0	Numerator for fractional time step
time_step_fract_den	1	Denominator for fractional time step. Example, if you want to use 60.3 sec as your time step, set time_step = 60, time_step_fract_num = 3, and time_step_fract_den = 10
max_dom	1	Number of domains (Nesting is not yet available for WRF-NMM, set max_dom=1.).
s_we (max_dom)	1	Start index in x (west-east) direction (leave as is)
e_we (max_dom)	124	End index in x (west-east) direction (staggered dimension)
s_sn (max_dom)	1	Start index in y (south-north) direction (leave as is)
e_sn (max_dom)	61	End index in y (south-north) direction (staggered dimension)
s_vert (max_dom)	1	Start index in z (vertical) direction (leave as is)
e_vert (max_dom)	61	End index in z (vertical) direction (staggered dimension). Note: This parameter refers to full levels including surface and top.

Variable Names	**Value (Example)**	**Description**
dx (max_dom)	.0534521	Grid length in x direction, in degrees for WRF-NMM.
dy (max_dom)	.0526316	Grid length in y direction, in degrees for WRF-NMM.
grid_id (max_dom)	1	Domain identifier.
tile_sz_x (max_dom)	0	Number of points in tile x direction.
tile_sz_y (max_dom)	0	Number of points in tile y direction.
numtiles (max_dom)	1	Number of tiles per patch (alternative to above two items).
nproc_x (max_dom)	-1	Number of processors in x–direction for decomposition.
nproc_y (max_dom)	-1	Number of processors in y-direction for decomposition: If -1: code will do automatic decomposition. If >1 for both: will be used for decomposition.
&physics		Physics options
chem_opt	0	Chemistry option - not yet available
mp_physics (max_dom)	5	Microphysics options: 0. no microphysics 1. Kessler scheme 2. Lin et al. scheme 3. WSM 3-class simple ice scheme 4. WSM 5-class scheme 5. Ferrier (Well-tested for WRF-NMM, used operationally at NCEP) 6. WSM 6-class graupel scheme. 8. Thompson et al. scheme. 98. NCEP 3-class simple ice scheme (to be removed) 99. NCEP 5-class scheme (to be removed)
ra_lw_physics	99	Long-wave radiation options: 0. No longwave radiation 1. RRTM scheme (Preliminarily tested for WRF-NMM.) 99. GFDL (Fels-Schwarzkopf) (Well-tested for WRF-NMM, used operationally at NCEP.)

Variable Names	Value (Example)	Description
ra_sw_physics	99	Short-wave radiation options: 0. No shortwave radiation 1. Dudhia scheme 2. Goddard short wave scheme 99. GFDL shortwave radiation scheme (Lacis-Hansen) (Well-tested for WRF-NMM, used operationally at NCEP.)
radt	60	Minutes between calls to the Dudhia and Goddard (GSFC) shortwave radiation schemes. Recommend 1 min per km of dx (e.g. 10 for 10 km).
nrads	100	**This flag is only for the WRF-NMM core**. Number of fundamental time steps between calls to GFDL shortwave radiation scheme (ra_sw_physics=99). NCEP's operational setting: “nrads” on the order of “3600/dt”. For more detailed results, use “1800/dt”.
nradl	100	**This flag is only for the WRF-NMM core**. Number of fundamental time steps between calls to GFDL longwave radiation scheme (ra_lw_physics=99). Can be set equal to “nrads”.
co2tf	1	**This flag is only for the WRF-NMM core.** Controls CO2 input used by the GFDL radiation scheme. 0: Read CO2 functions data from pre-generated file 1: Generate CO2 functions data internally
sf_sfclay_physics	2	Surface-layer options: 0. No surface-layer scheme 1. Monin-Obukhov scheme 2. Janjic scheme (Well-tested for WRF-NMM, used operationally at NCEP) 3. NCEP Global Forecast System scheme (Well-tested by NCEP for WRF-NMM)

Variable Names	**Value (Example)**	**Description**
sf_surface_physics	99	Land-surface options: 0. No surface temperature prediction 1. Thermal diffusion scheme 2. Noah Land-Surface Model 3. RUC Land-Surface Model 99. NMM Land Surface Model (Well-tested for WRF-NMM, used operationally at NCEP.)
bl_pbl_physics	2	Boundary-layer options: 0. No boundary-layer 1. YSU scheme (Preliminarily tested for WRF-NMM.) 2. Mellor-Yamada-Janjic TKE scheme (Well-tested for WRF-NMM, used operationally at NCEP) 3. NCEP Global Forecast System scheme (Well-tested by NCEP for WRF-NMM) 99. MRF scheme (to be removed)
bldt	3	**This flag is only for WRF-ARW core**. Minutes between boundary-layer physics calls.
nphs (max_dom)	10	**This flag is only for WRF-NMM core**. Number of fundamental time steps between calls to turbulence and microphysics
cu_physics	2	Cumulus scheme options: 0. No cumulus scheme (Well-tested for WRF-NMM) 1. Kain-Fritsch scheme (Preliminarily tested for WRF-NMM) 2. Betts-Miller-Janjic scheme (Well-tested for WRF-NMM, used operationally at NCEP) 3. Grell-Devenyi ensemble scheme 4. Simplified Arakawa-Schubert scheme (Well-tested for WRF-NMM by NCEP) 99. Older version of Kain-Fritsch scheme
cudt	3	**This flag is only for WRF-ARW core**. Minutes between cumulus physics calls.
ncnvc (max_dom)	10	**This flag is only for WRF-NMM core**. Number of fundamental time steps between calls to convection. ***Note that "ncnvc" should be set equal to "nphs".***

Variable Names	**Value (Example)**	**Description**
isfflx	0	Heat and moisture fluxes from the surface for the “Monin-Obukhov scheme” (sf_sfclay_physics=1): 0. No flux from the surface 1. With fluxes from the surface
ifsnow	0	Snow-cover effects for “Thermal Diffusion scheme” (sf_surface_physics=1): 0. No snow-cover effect 1. With snow-cover effect
icloud	0	Cloud effect to the optical depth in the Dudhia shortwave (ra_sw_physics=1) and RRTM longwave radiation (ra_lw_physics=1) schemes. 0. No cloud effect 1. With cloud effect
num_soil_layers	4	Number of soil layers in land surface model. Options available: 4. (for NMM and NOAH-LSM) (Well-tested for WRF-NMM, used operationally at NCEP) 5. Thermal diffusion scheme 6. RUC Land Surface Model
mp_zero_out	0	For non-zero mp_physics options, to keep water vapor positive (Qv >= 0), and to set the other moisture fields smaller than a threshold value to zero. 0. No action is taken, no adjustment to any moist field. (conservation maintained) 1. All moist arrays, except for Qv, are set to zero if they fall below a critical value. (No conservation) 2. Qv<0 are set to zero, and all other moist arrays that fall below the critical value defined in the flag “mp_zero_out_thresh” are set to zero. (No conservation.) ***For WRF-NMM, mp_zero_out MUST BE set to 0.***

Variable Names	**Value (Example)**	**Description**
mp_zero_out_thresh	1.e-8	Critical value for moisture variable threshold, below which moist arrays (except for Qv) are set to zero (unit: kg/kg). Default value is “1.e-8”.
&dynamics		Dynamics options:
dyn_opt	4	4. WRF-NMM dynamics
rk_ord	3	**This flag is only for WRF-ARW core**
diff_opt	0	**This flag is only for WRF-ARW core**
km_opt	1	**This flag is only for WRF-ARW core**
damp_opt	1	**This flag is only for WRF-ARW core**
zdamp	5000	**This flag is only for WRF-ARW core**
dampcoef	0.01	**This flag is only for WRF-ARW core**
khdif	0	**This flag is only for WRF-ARW core**
kvdif	0	**This flag is only for WRF-ARW core**
mix_cr_len	200	**This flag is only for WRF-ARW core**
smdiv	0.1	**This flag is only for WRF-ARW core**
emdiv	0.01	**This flag is only for WRF-ARW core**
epssm	0.1	**This flag is only for WRF-ARW core**
time_step_sound	4	**This flag is only for WRF-ARW core**
non_hydrostatic	.true.	Whether running the model in hydrostatic or non-hydrostatic model.
&bc_control		Boundary condition control.
spec_bdy_width	1	Total number of rows for specified boundary value nudging. ***It MUST be set to 1 for WRF-NMM core.***
spec_zone	1	**This flag is only for WRF-ARW core**
relax_zone	4	**This flag is only for WRF-ARW core**
specified (max_dom)	.true.	**This flag is only for WRF-ARW core**
periodic_x (max_dom)	.false.	**This flag is only for WRF-ARW core**
symmetric_xs (max_dom)	.false.	**This flag is only for WRF-ARW core**
symmetric_xe (max_dom)	.false.	**This flag is only for WRF-ARW core**

Variable Names	Value (Example)	Description
open_xs (max_dom)	.false.	**This flag is only for WRF-ARW core**
open_xe (max_dom)	.false.	**This flag is only for WRF-ARW core**
periodic_y (max_dom)	.false.	**This flag is only for WRF-ARW core**
symmetric_ys (max_dom)	.false.	**This flag is only for WRF-ARW core**
symmetric_ye (max_dom)	.false.	**This flag is only for WRF-ARW core**
open_ys (max_dom)	.false.	**This flag is only for WRF-ARW core**
open_ye (max_dom)	.false.	**This flag is only for WRF-ARW core**
nested (max_dom)	.false.	**This flag is only for WRF-ARW core**
&namelist_quilt		Option for asynchronized I/O for MPI applications.
nio_tasks_per_group	0	Default value is 0, means no quilting; value > 0 quilting I/O
nio_groups	1	Default is 1, do NOT change.

Software requirement

- FORTRAN 90 or 95 and C compilers
- Perl 5.04 or higher versions
- If MPI or OpenMP compilation is desired, requires MPI or OpenMP libraries
- WRF I/O API supports netCDF, PHD5 and GriB-1 formats, hence one of these libraries needs to be available on the computer used to compile and run WRF.

Before You Start

Before configuring and compiling the WRF-NMM code, the following points should be checked:

1. Verify netCDF is installed. NetCDF software can be acquired from UNIDATA at: *http://my.unidata.ucar.edu/content/software/netcdf/index.html*

2. Make sure netCDF is installed either in /usr/local or the path to the ***netCDF*** libraries and its include/ directory is defined by the environmental variable NETCDF. For example,

```
setenv NETCDF /path-to-netcdf-library
```

(On NCAR IBM supercomputers, it is not necessary to set the NETCDF environment variable because it will be automatically created during the ***"./configure"*** process. If the configuration is successful, a "netcdf_links" sub-directory should be created in WRFV2 main directory.)

For LINUX Platforms:

- A helpful guide to building WRFV2 using PGI 5.2-2 compilers on a 32-bit or 64-bit LINUX system can be found at: *http://www.pgroup.com/resources/wrf/wrfv2_pgi52.htm*.
- MPICH for LINUX-PCs can be downloaded from: *http://www-unix.mcs.anl.gov/mpi/mpich*.
- NetCDF and WRF must be compiled using the same compiler. The netCDF library compiled with PGI is usually located in ***/usr/local/netcdf-pgi***

Path names for the compilers and libraries listed above should be defined in the shell configuration files (such as **".cshrc"** and **".login"**). For example:

```
set path = (    /usr/pgi/bin /usr/pgi/lib /usr/local/ncarg/bin \
                /usr/local/mpich-pgi /usr/local/mpich-pgi/bin \
                /usr/local/netcdf-pgi/bin /usr/local/netcdf/include)
setenv PGI /usr/pgi
setenv NETCDF /usr/local/netcdf-pgi
setenv LM_LICENSE_FILE $PGI/license.dat
setenv LD_LIBRARY_PATH /usr/lib:/usr/local/lib:/usr/pgi/linux86/lib:/usr/local/netcdf-
pgi/lib
```

How to Obtain and Open WRFV2 Package?

WRF-NMM source code ***tar*** file may be downloaded from:
http://www.dtcenter.org/wrf-nmm/users
Once the ***tar*** file is obtained, ***gunzip*** and ***untar*** the file. The end product will be a ***WRFV2/*** directory that contains:

Makefile	Top-level makefile
README	General information about WRF code
README.NMM	NMM specific information
README_test_cases	Explanation of the test cases for WRF-ARW
Registry/	Directory for WRF Registry file
arch/	Directory where compile options are gathered
chem/	Directory for chemistry modules

clean	script to clean created files and executables
compile	script for compiling WRF code
configure	script to configure the configure.wrf file for compile
dyn_em	Directory for WRF-ARW dynamic modules
dyn_exp/	Directory for a 'toy' dynamic core
dyn_nmm/	Directory for WRF-NMM dynamic modules
external/	Directory that contains external packages, such as those for IO, time keeping and MPI
frame/	Directory that contains modules for WRF framework
inc/	Directory that contains include files
main/	Directory for main routines, such as wrf.F, and all executables
phys/	Directory for all physics modules
run/	Directory where one may run WRF
share/	Directory that contains mostly modules for WRF mediation layer and WRF I/O
test/	Directory containing sub-directories where one may run specific configurations of WRF. test case Only *nmm_real* is relevant to WRF-NMM
tools/	Directory that contains tools

How to Configure WRFV2?

WRF-NMM has been tested on the following platforms:

Vendor	Hardware	O.S.	Compiler
IBM	SP Power-*x*	AIX	vendor
SGI	MIPS	IRIX	vendor
HP/COMPAQ/DEC	Alpha	Tru64	vendor
Various	IA-32	LINUX	PGI
Various	Opteron	LINUX	PGI

To configure WRF, go to the WRFV2 (top) directory (***cd WRFV2***) and type:

./configure

You will be given a list of choices for your computer. These choices range from compiling for a single processor job, to using OpenMP shared-memory (SM) or distributed-memory (DM) parallelization options for multiple processors. Some options support nesting (currently not available for WRF-NMM), others do not.

<u>Choices for IBM machines are as follows:</u>

```
 1. AIX                    (single-threaded, no nesting)
```

2. AIX SM (OpenMP, no nesting)
3. AIX DM-Parallel (RSL_LITE, IBM-MPI, allows nesting)
4. AIX DM-Parallel (RSL, IBM-MPI, allows nesting)
5. AIX DM-Parallel (RSL, IBM-MPI, allows nesting) (PARALLEL HDF5)
6. AIX DM-Parallel (RSL_LITE, IBM-MPI, allows nesting) (PARALLEL HDF5)
7. AIX DM-Parallel/SM-Parallel (not recommended)
   (RSL, IBM-MPI, OpenMP, allows nesting)
8. AIX DM-Parallel (RSL, IBM-MPI, MCEL, May 2003, experimental)
9. AIX DM-Parallel ESMF (RSL, IBM-MPI, ESMF coupling, no nesting, experimental)
10. AIX (Single-threaded, nesting using RSL without MPI)
11. AIX (OpenMP, nesting using RSL without MPI)

For WRF-NMM V2 on IBM systems, option 4 is recommended.

Choices for LINUX operating systems are as follows:

1. PC Linux i486 i586 i686, PGI compiler (Single-threaded, no nesting)
2. PC Linux i486 i586 i686, PGI compiler
   (Single threaded, allows nesting using RSL without MPI)
3. PC Linux i486 i586 i686, PGI compiler SM-Parallel (OpenMP, no nesting)
4. PC Linux i486 i586 i686, PGI compiler SM-Parallel
   (OpenMP, allows nesting using RSL without MPI)
5. PC Linux i486 i586 i686, PGI compiler DM-Parallel (RSL, MPICH, allows nesting)
6. PC Linux i486 i586 i686, PGI compiler DM-Parallel
   (RSL_LITE, MPICH, allows nesting)
7. Intel xeon i686 ia32 Xeon Linux, ifort compiler (Single-threaded, no nesting)
8. Intel xeon i686 ia32 Xeon Linux, ifort compiler
   (Single threaded, allows nesting using RSL without MPI)
9. Intel xeon i686 ia32 Xeon Linux, ifort compiler (OpenMP)
10. Intel xeon i686 ia32 Xeon Linux, ifort compiler SM-Parallel
    (OpenMP, allows nesting using RSL without MPI)
11. Intel xeon i686 ia32 Xeon Linux, ifort+icc compiler DM-Parallel
    (RSL, MPICH, allows nesting)
12. Intel xeon i686 ia32 Xeon Linux, ifort+gcc compiler DM-Parallel
    (RSL, MPICH, allows nesting)
13. PC Linux i486 i586 i686, PGI compiler, ESMF
    (Single-threaded, ESMF coupling, no nesting)

For WRF-NMM V2 on LINUX operating systems, option 5 is recommended.

Check the ***configure.wrf*** file provided; and edit for compile options/paths, if necessary.

How to Compile WRFV2 for NMM core?

To compile WRFV2 for the NMM dynamic core, the following environment variable must be set:

```
setenv WRF_NMM_CORE 1
```

Once this environment variable is set, enter the following command:

```
./compile nmm_real
```

Note that entering:

```
./compile -h
```
or
```
./compile
```

produces a listing of all of the available compile options (only ***nmm_real*** is relevant to the WRF-NMM core).

To remove all object and executable files, type: ***clean***

To remove all built files, including ***configure.wrf,*** type: ***clean –a***. This action is recommended if a mistake is made during the installation process, or if the ***Registry.NMM file*** is edited.

When the compilation is successful, two executables are created in the **main/** directory:

real_nmm.exe and ***wrf.exe***.
real_nmm.exe: WRF-NMM initialization
wrf.exe: WRF-NMM model integration

These executables are linked to the ***run*** and ***test/nmm_real***irectories. The ***test/nmm_real/*** and ***run/*** directories are working directories used for actually running the model.

How to Run WRF for NMM Core?

Running a real-data case requires first successfully running the WRF-NMM Standard Initialization (SI) program (See Chapter 3 for a description of the WRF-NMM SI and directions for installing and running this package).

Running *real_nmm.exe*:

1. Change to the working directory of choice (***cd test/nmm_real*** or ***cd run***).

2. Make sure the files listed below reside in or are linked to the working-directory chosen to run the model:

ETAMPNEW_DATA	(WRFV2/run)
GENPARM.TBL	(WRFV2/run)
gribmap.txt	(WRFV2/run)

LANDUSE.TBL	(WRFV2/run)
namelist.input	(WRFV2/test/nmm_real)
real_nmm.exe	(WRFV2/run)
RRTM_DATA	(WRFV2/run)
SOILPARM.TBL	(WRFV2/run)
tr49t67	(WRFV2/run)
tr49t85	(WRFV2/run)
tr67t85	(WRFV2/run)
VEGPARM.TBL	(WRFV2/run)
wrf.exe	(WRFV2/run)

3. Make sure the ***wrf_real_input_nm.**** files from the WRF-NMM SI either reside in or are linked to the working directory chosen to run the model.

4. Edit the namelist.input file in the working directory for dates, domain size, time step, output options, and physics options (see Description of Namelist Variables section for details).

5. The command issued to run ***"real_nmm.exe"*** in the working directory will depend on the operating system.

On LINUX-MPI systems, the command is:

```
mpirun -np n real_nmm.exe
```

where **"n"** defines the number of processors to use. For single processor PCs, use 1.

For batch jobs on IBM systems, the command is:

```
poe real_nmm.exe
```

For interactive runs on IBMs, the command is:

```
poe real_nmm.exe -rmpool 1 -procs n
```

where ***"n"*** stands for the number of processors (CPUs) to be used.
When "***real_nmm.exe"*** is successful, the following files that are used by ***wrf.exe*** should be found in the working-directory:

wrfinput_d01	(Initial conditions, single time level data.)
wrfbdy_d01	(Boundary conditions data for multiple time steps.)

To check whether the run is successful, look for "SUCCESS COMPLETE REAL_NMM INIT" at the end of "***rsl.out.0000***"

Running “wrf.exe”:

Note: Running ***“wrf.exe”*** requires a successful run of “***real_nmm.exe”*** as explained above.

1. If the working directory used to run “***wrf.exe”*** is different than the one used to run “***real_nmm.exe***”, make sure ***wrfinput_d01*** and ***wrfbdy_d01, as well as the files listed above in the real_nmm.exe discussion,*** are in your working-directory (you may link the files to this directory).

2. The command issued to run ***“wrf.exe”*** in the working directory will depend on the operating system:

   On LINUX-MPI systems, the command is:

   ```
   mpirun -np n wrf.exe
   ```

   where ***“n”*** defines the number of processors to use. For single processor PCs, use 1.

   For batch jobs on IBM systems, the command is:

   ```
   poe wrf.exe
   ```

   For interactive runs on IBMs, the command is:

   ```
   poe wrf.exe -rmpool 1 -procs n
   ```

   where ***“n”*** stands for number of processors (CPUs) to be used.

A successful run of ***“wrf.exe”*** will produce output files with the following naming convention:

wrfout_d01_yyyy-mm-dd_hh:mm:ss

For example, the first output file for a run started at 0000 UTC, 23rd January 2005 would be:

wrfout_d01_2005-01-23_00:00:00

To check whether the run is successful, look for “SUCCESS COMPLETE WRF” at the end of ***rsl.out.0000.***

The times written to an output file can be checked by typing:

```
ncdump -v Times wrfout_d01_2005-01-23_00:00:00
```

The number of "***wrfout***" files generated by a successful run of ***"wrf.exe"*** and the number of output times per "***wrfout***" file will depend on the output options specified in "***namelist.input***" (i.e***.***, ***frames_per_outfile*** and ***history interval***).

Restart files can also be created, if restart frequency (***restart_interval*** in ***namelist.input***) is set within the total integration length. Restart files have the following naming convention:

wrfrst_d01_yyyy-mm-dd_hh:mm:ss

Real Data Test Case: 2005 January 23/00 through 24/00

The above described steps can be tested on the real data set provided. The test data set is accessible from the WRF-NMM download page. Under "WRF Model Test Data (regenerated for V2.1.1 WRF-NMM)", select the January data. This is a 55x91, 15-km domain centered over the eastern US.

- After running the ***real_nmm.exe*** program, the files ***wrfinput_d01*** and ***wrfbdy_d01***, should appear in the working directory. These files will be used by the WRF model.
- The ***wrf.exe*** program is executed next. This step should take a few minutes (only a 24 h forecast is requested in the ***namelist***),
- The output file ***wrfout_d01:2005-01-23_00:00:00*** should contain a 24 h forecast at 1 h intervals.

List of Fields in WRF-NMM Output

The following is edited output from the netCDF command '***ncdump***':

ncdump -h wrfout_d01_yyyy_mm_dd-hh:mm:ss

An example:

ncdump –h wrfout_d01_2005-01-23_00:00:00

```
dimensions:
        Time = UNLIMITED ; // (1 currently)
        DateStrLen = 19 ;
        west_east = 123 ;
        south_north = 201 ;
        bottom_top = 60 ;
        ext_scalar = 1 ;
        soil_layers_stag = 4 ;
        bottom_top_stag = 61 ;
variables:
        char Times(Time, DateStrLen) ;
        float LU_INDEX(Time, south_north, west_east) ;
```

```
int LMH(Time, south_north, west_east) ;
int LMV(Time, south_north, west_east) ;
float HBM2(Time, south_north, west_east) ;
float HBM3(Time, south_north, west_east) ;
float VBM2(Time, south_north, west_east) ;
float VBM3(Time, south_north, west_east) ;
float SM(Time, south_north, west_east) ;
float SICE(Time, south_north, west_east) ;
float HTM(Time, bottom_top, south_north, west_east) ;
float VTM(Time, bottom_top, south_north, west_east) ;
float PD(Time, south_north, west_east) ;
float FIS(Time, south_north, west_east) ;
float RES(Time, south_north, west_east) ;
float T(Time, bottom_top, south_north, west_east) ;
float Q(Time, bottom_top, south_north, west_east) ;
float U(Time, bottom_top, south_north, west_east) ;
float V(Time, bottom_top, south_north, west_east) ;
float DX_NMM(Time, south_north, west_east) ;
float PDTOP(Time, ext_scalar) ;
float PT(Time, ext_scalar) ;
float PBLH(Time, south_north, west_east) ;
float USTAR(Time, south_north, west_east) ;
float Z0(Time, south_north, west_east) ;
float THS(Time, south_north, west_east) ;
float QS(Time, south_north, west_east) ;
float TWBS(Time, south_north, west_east) ;
float QWBS(Time, south_north, west_east) ;
float PREC(Time, south_north, west_east) ;
float APREC(Time, south_north, west_east) ;
float ACPREC(Time, south_north, west_east) ;
float CUPREC(Time, south_north, west_east) ;
float SNO(Time, south_north, west_east) ;
float SI(Time, south_north, west_east) ;
float CLDEFI(Time, south_north, west_east) ;
float TH10(Time, south_north, west_east) ;
float Q10(Time, south_north, west_east) ;
float PSHLTR(Time, south_north, west_east) ;
float TSHLTR(Time, south_north, west_east) ;
float QSHLTR(Time, south_north, west_east) ;
float Q2(Time, bottom_top, south_north, west_east) ;
float AKHS_OUT(Time, south_north, west_east) ;
float ALBASE(Time, south_north, west_east) ;
float ALBEDO(Time, south_north, west_east) ;
float CNVBOT(Time, south_north, west_east) ;
float CNVTOP(Time, south_north, west_east) ;
float CZEN(Time, south_north, west_east) ;
```

```
float CZMEAN(Time, south_north, west_east) ;
float GLAT(Time, south_north, west_east) ;
float GLON(Time, south_north, west_east) ;
float RADOT(Time, south_north, west_east) ;
float SIGT4(Time, south_north, west_east) ;
float TGROUND(Time, south_north, west_east) ;
float CWM(Time, bottom_top, south_north, west_east) ;
float F_ICE(Time, bottom_top, south_north, west_east) ;
float F_RAIN(Time, bottom_top, south_north, west_east) ;
float F_RIMEF(Time, bottom_top, south_north, west_east) ;
float SR(Time, south_north, west_east) ;
float CFRACH(Time, south_north, west_east) ;
float CFRACL(Time, south_north, west_east) ;
float CFRACM(Time, south_north, west_east) ;
int ISLOPE(Time, south_north, west_east) ;
float SLDPTH(Time, bottom_top) ;
float CMC(Time, south_north, west_east) ;
float GRNFLX(Time, south_north, west_east) ;
float PCTSNO(Time, south_north, west_east) ;
float SOILTB(Time, south_north, west_east) ;
float VEGFRC(Time, south_north, west_east) ;
float SH2O(Time, soil_layers_stag, south_north, west_east) ;
float SMC(Time, soil_layers_stag, south_north, west_east) ;
float STC(Time, soil_layers_stag, south_north, west_east) ;
float PINT(Time, bottom_top_stag, south_north, west_east) ;
float W(Time, bottom_top_stag, south_north, west_east) ;
float ACFRCV(Time, south_north, west_east) ;
float ACFRST(Time, south_north, west_east) ;
float SSROFF(Time, south_north, west_east) ;
float BGROFF(Time, south_north, west_east) ;
float RLWIN(Time, south_north, west_east) ;
float ALWIN(Time, south_north, west_east) ;
float ALWOUT(Time, south_north, west_east) ;
float ALWTOA(Time, south_north, west_east) ;
float RSWIN(Time, south_north, west_east) ;
float RSWOUT(Time, south_north, west_east) ;
float ASWIN(Time, south_north, west_east) ;
float ASWOUT(Time, south_north, west_east) ;
float ASWTOA(Time, south_north, west_east) ;
float SFCSHX(Time, south_north, west_east) ;
float SFCLHX(Time, south_north, west_east) ;
float SUBSHX(Time, south_north, west_east) ;
float SNOPCX(Time, south_north, west_east) ;
float SFCUVX(Time, south_north, west_east) ;
float POTEVP(Time, south_north, west_east) ;
float RLWTT(Time, bottom_top, south_north, west_east) ;
```

```
float RSWTT(Time, bottom_top, south_north, west_east) ;
float TCUCN(Time, bottom_top, south_north, west_east) ;
float TRAIN(Time, bottom_top, south_north, west_east) ;
int NCFRCV(Time, south_north, west_east) ;
int NCFRST(Time, south_north, west_east) ;
int NPHS0(Time, ext_scalar) ;
int NPREC(Time, ext_scalar) ;
int NCLOD(Time, ext_scalar) ;
int NHEAT(Time, ext_scalar) ;
int NRDLW(Time, ext_scalar) ;
int NRDSW(Time, ext_scalar) ;
int NSRFC(Time, ext_scalar) ;
float AVRAIN(Time, ext_scalar) ;
float AVCNVC(Time, ext_scalar) ;
float ARDLW(Time, ext_scalar) ;
float ARDSW(Time, ext_scalar) ;
float ASRFC(Time, ext_scalar) ;
float LANDMASK(Time, south_north, west_east) ;
float SMOIS(Time, soil_layers_stag, south_north, west_east) ;
float PSFC(Time, south_north, west_east) ;
float TH2(Time, south_north, west_east) ;
float U10(Time, south_north, west_east) ;
float V10(Time, south_north, west_east) ;
float SMSTAV(Time, south_north, west_east) ;
float SMSTOT(Time, south_north, west_east) ;
float SFROFF(Time, south_north, west_east) ;
float UDROFF(Time, south_north, west_east) ;
int IVGTYP(Time, south_north, west_east) ;
int ISLTYP(Time, south_north, west_east) ;
float VEGFRA(Time, south_north, west_east) ;
float SFCEVP(Time, south_north, west_east) ;
float GRDFLX(Time, south_north, west_east) ;
float ACSNOW(Time, south_north, west_east) ;
float ACSNOM(Time, south_north, west_east) ;
float SNOW(Time, south_north, west_east) ;
float CANWAT(Time, south_north, west_east) ;
float SST(Time, south_north, west_east) ;
float WEASD(Time, south_north, west_east) ;
float TKE_MYJ(Time, bottom_top, south_north, west_east) ;
float EL_MYJ(Time, bottom_top, south_north, west_east) ;
float EXCH_H(Time, bottom_top, south_north, west_east) ;
float THZ0(Time, south_north, west_east) ;
float QZ0(Time, south_north, west_east) ;
float UZ0(Time, south_north, west_east) ;
float VZ0(Time, south_north, west_east) ;
float QSFC(Time, south_north, west_east) ;
```

```
          float HTOP(Time, south_north, west_east) ;
          float HBOT(Time, south_north, west_east) ;
          float HTOPD(Time, south_north, west_east) ;
          float HBOTD(Time, south_north, west_east) ;
          float HTOPS(Time, south_north, west_east) ;
          float HBOTS(Time, south_north, west_east) ;
          float CUPPT(Time, south_north, west_east) ;
          float SNOWH(Time, south_north, west_east) ;
          float SMFR3D(Time, soil_layers_stag, south_north, west_east) ;
          int ITIMESTEP(Time, ext_scalar) ;

Global attributes:
                    :TITLE = " OUTPUT FROM WRF V2.0.3.1 MODEL" ;
                    :START_DATE = "2005-04-27_00:00:00" ;
                    :SIMULATION_START_DATE = "2005-04-27_00:00:00" ;
                    :WEST-EAST_GRID_DIMENSION = 124 ;
                    :SOUTH-NORTH_GRID_DIMENSION = 202 ;
                    :BOTTOM-TOP_GRID_DIMENSION = 61 ;
                    :GRIDTYPE = "E" ;
                    :DYN_OPT = 4 ;
                    :DIFF_OPT = 0 ;
                    :KM_OPT = 1 ;
                    :DAMP_OPT = 1 ;
                    :KHDIF = 0.f ;
                    :KVDIF = 0.f ;
                    :MP_PHYSICS = 5 ;
                    :RA_LW_PHYSICS = 99 ;
                    :RA_SW_PHYSICS = 99 ;
                    :SF_SFCLAY_PHYSICS = 2 ;
                    :SF_SURFACE_PHYSICS = 99 ;
                    :BL_PBL_PHYSICS = 2 ;
                    :CU_PHYSICS = 2 ;
                    :WEST-EAST_PATCH_START_UNSTAG = 1 ;
                    :WEST-EAST_PATCH_END_UNSTAG = 123 ;
                    :WEST-EAST_PATCH_START_STAG = 1 ;
                    :WEST-EAST_PATCH_END_STAG = 124 ;
                    :SOUTH-NORTH_PATCH_START_UNSTAG = 1 ;
                    :SOUTH-NORTH_PATCH_END_UNSTAG = 201 ;
                    :SOUTH-NORTH_PATCH_START_STAG = 1 ;
                    :SOUTH-NORTH_PATCH_END_STAG = 202 ;
                    :BOTTOM-TOP_PATCH_START_UNSTAG = 1 ;
                    :BOTTOM-TOP_PATCH_END_UNSTAG = 60 ;
                    :BOTTOM-TOP_PATCH_START_STAG = 1 ;
                    :BOTTOM-TOP_PATCH_END_STAG = 61 ;
                    :DX = 0.0534521f ;
                    :DY = 0.0526316f ;
```

```
:DT = 18.f ;
:CEN_LAT = 40.f ;
:CEN_LON = -115.f ;
:TRUELAT1 = 40.f ;
:TRUELAT2 = 0.f ;
:MOAD_CEN_LAT = 0.f ;
:STAND_LON = 0.f ;
:GMT = 0.f ;
:JULYR = 2005 ;
:JULDAY = 117 ;
:MAP_PROJ = 203 ;
:MMINLU = "USGS" ;
:ISWATER = 16 ;
:ISICE = 24 ;
:ISURBAN = 1 ;
:ISOILWATER = 14 ;
:I_PARENT_START = 1 ;
:J_PARENT_START = 1 ;
```

Extended Reference List for WRF-NMM Dynamics and Physics

Arakawa, A., and W. H. Schubert, 1974: Interaction of a cumulus cloud ensemble with the large scale environment. Part I. J. Atmos. Sci., 31, 674-701.

Chen, F., Z. Janjic and K. Mitchell, 1997: Impact of atmospheric surface-layer parameterization in the new land-surface scheme of the NCEP mesoscale Eta model. *Boundary-Layer Meteorology*, 48

Chen, S.-H., and W.-Y. Sun, 2002: A one-dimensional time dependent cloud model. J. Meteor. Soc. Japan, 80, 99–118.

Chen, F., and J. Dudhia, 2001: Coupling an advanced land-surface/ hydrology model with the Penn State/ NCAR MM5 modeling system. Part I: Model description and implementation. Mon. Wea. Rev., 129, 569–585.

Chou M.-D., and M. J. Suarez, 1994: An efficient thermal infrared radiation parameterization for use in general circulation models. NASA Tech. Memo. 104606, 3, 85pp.

Dudhia, J., 1989: Numerical study of convection observed during the winter monsoon experiment using a mesoscale two-dimensional model, J. Atmos. Sci., 46, 3077–3107.

Ek, M. B., K. E. Mitchell, Y. Lin, E. Rogers, P. Grunmann, V. Koren, G. Gayno, and J. D. Tarpley, 2003: Implementation of NOAH land surface model advances in the NCEP operational mesoscale Eta model. J. Geophys. Res., 108, No. D22, 8851, do1:10.1029/2002JD003296.

Fels, S. B., and M. D. Schwarzkopf, 1975: The simplified exchange approximation: A new method for radiative transfer calculations. J. Atmos. Sci., 32, 1475-1488.

Ferrier, B. S., Y. Lin, T. Black, E. Rogers, and G. DiMego, 2002: Implementation of a new grid-scale cloud and precipitation scheme in the NCEP Eta model. Preprints, 15th Conference on Numerical Weather Prediction, San Antonio, TX, Amer. Meteor. Soc., 280-283.

Grell, G. A., 1993: Prognostic Evaluation of Assumptions Used by Cumulus Parameterizations. Mon. Wea. Rev., 121, 764-787.

Grell, G. A., and D. Devenyi, 2002: A generalized approach to parameterizing convection combining ensemble and data assimilation techniques. Geophys. Res. Lett., 29(14), Article 1693.

Hong, S.-Y., J. Dudhia, and S.-H. Chen, 2004: A Revised Approach to Ice Microphysical Processes for the Bulk Parameterization of Clouds and Precipitation, Mon. Wea. Rev., 132, 103–120.

Hong, S.-Y., H.-M. H. Juang, and Q. Zhao, 1998: Implementation of prognostic cloud scheme for a regional spectral model, Mon. Wea. Rev., 126, 2621–2639.

Hong, S.-Y., and H.-L. Pan, 1996: Nonlocal boundary layer vertical diffusion in a medium-range forecast model, Mon. Wea. Rev., 124, 2322–2339.

Janjic, Z. I., 1984: Non–linear advection schemes and energy cascade on semi–staggered grids. *Monthly Weather Review*, 112, 1234–1245.

Janjic, Z. I., 1990: The step–mountain coordinates: physical package. *Monthly Weather Review*, 118, 1429–1443.

Janjic, Z. I., 1994: The step–mountain eta coordinate model: further developments of the convection, viscous sublayer and turbulence closure schemes. *Monthly Weather Review*, 122, 927–945.

Janjic, Z. I., 1996a: The Mellor-Yamada level 2.5 scheme in the NCEP Eta Model. 11th Conference on Numerical Weather Prediction, Norfolk, VA, 19-23 August 1996; American Meteorological Society, Boston, MA, 333-334.

Janjic, Z. I., 1996b: The Surface Layer in the NCEP Eta Model. 11th Conf. on NWP, Norfolk, VA, American Meteorological Society, 354–355.

Janjic, Z. I., 1997: Advection Scheme for Passive Substances in the NCEP Eta Model. Research Activities in Atmospheric and Oceanic Modelling, WMO, Geneva, CAS/JSC WGNE, 3.14.

Janjic, Z. I., 2000: Comments on “Development and Evaluation of a Convection Scheme for Use in Climate Models. *Journal of the Atmospheric Sciences*, 57, p. 3686

Janjic, Z. I., 2001: Nonsingular Implementation of the Mellor-Yamada Level 2.5 Scheme in the NCEP Meso model. NCEP Office Note No. 437, 61 pp.

Janjic, Z. I., 2002a: A Nonhydrostatic Model Based on a New Approach. EGS XVIII, Nice France, 21-26 April 2002.

Janjic, Z. I., 2002b: Nonsingular Implementation of the Mellor–Yamada Level 2.5 Scheme in the NCEP Meso model, NCEP Office Note, No. 437, 61 pp.

Janjic, Z. I., 2003a: A Nonhydrostatic Model Based on a New Approach. *Meteorology and Atmospheric Physics*, 82, 271-285. (Online: http://dx.doi.org/10.1007/s00703-001-0587-6).

Janjic, Z. I., 2003b: The NCEP WRF Core and Further Development of Its Physical Package. 5th International SRNWP Workshop on Non-Hydrostatic Modelling, Bad Orb, Germany, 27-29 October.

Janjic, Z. I., 2004: The NCEP WRF Core. 12.7, Extended Abstract, 20th Conference on Weather Analysis and Forecasting/16th Conference on Numerical Weather Prediction, Seattle, WA, American Meteorological Society.

Janjic, Z. I., J. P. Gerrity, Jr. and S. Nickovic, 2001: An Alternative Approach to Nonhydrostatic Modeling. *Monthly Weather Review*, 129, 1164-1178.

Janjic, Z. I., T. L. Black, E. Rogers, H. Chuang and G. DiMego, 2003: The NCEP Nonhydrostatic Meso Model (NMM) and First Experiences with Its Applications. EGS/EGU/AGU Joint Assembly, Nice, France, 6-11 April.

Janjic, Z. I, T. L. Black, E. Rogers, H. Chuang and G. DiMego, 2003: The NCEP Nonhydrostatic Mesoscale Forecasting Model. 12.1, Extended Abstract, 10th Conference on Mesoscale Processes, Portland, OR, American Meteorological Society. (Available Online).

Kain J. S. and J. M. Fritsch, 1990: A One-Dimensional Entraining/Detraining Plume Model and Its Application in Convective Parameterization. Journal of the Atmospheric Sciences, 47, No. 23, pp. 2784–2802.

Kain, J. S., and J. M. Fritsch, 1993: Convective parameterization for mesoscale models: The Kain-Fritcsh scheme, the representation of cumulus convection in numerical models, K. A. Emanuel and D.J. Raymond, Eds., Amer. Meteor. Soc., 246 pp.

Kain J. S., 2004: The Kain–Fritsch Convective Parameterization: An Update. Journal of Applied Meteorology, 43, No. 1, pp. 170–181.

Kessler, E., 1969: On the distribution and continuity of water substance in atmospheric circulation, Meteor. Monogr., 32, Amer. Meteor. Soc., 84 pp.

Lacis, A. A., and J. E. Hansen, 1974: A parameterization for the absorption of solar radiation in the earth's atmosphere. J. Atmos. Sci., 31, 118–133.

Lin, Y.-L., R. D. Farley, and H. D. Orville, 1983: Bulk parameterization of the snow field in a cloud model. J. Climate Appl. Meteor., 22, 1065–1092.

Miyakoda, K., and J. Sirutis, 1986: Manual of the E-physics. [Available from Geophysical Fluid Dynamics Laboratory, Princeton University, P.O. Box 308, Princeton, NJ 08542]

Mlawer, E. J., S. J. Taubman, P. D. Brown, M. J. Iacono, and S. A. Clough, 1997: Radiative transfer for inhomogeneous atmosphere: RRTM, a validated correlated-k model for the longwave. J. Geophys. Res., 102 (D14), 16663–16682.

Pan, H.-L. and W.-S. Wu, 1995: Implementing a Mass Flux Convection Parameterization Package for the NMC Medium-Range Forecast Model. NMC Office Note, No. 409, 40pp. [Available from NCEP/EMC, W/NP2 Room 207, WWB, 5200 Auth Road, Washington, DC 20746-4304]

Pan, H-L. and L. Mahrt, 1987: Interaction between soil hydrology and boundary layer developments. Boundary Layer Meteor., 38, 185-202.

Rutledge, S. A., and P. V. Hobbs, 1984: The mesoscale and microscale structure and organization of clouds and precipitation in midlatitude cyclones. XII: A diagnostic modeling study of precipitation development in narrow cloud-frontal rainbands. J. Atmos. Sci., 20, 2949–2972.

Sadourny. R., 1975: The Dynamics of Finite-Difference Models of the Shallow-Water Equations. *Journal of the Atmospheric Sciences*: Vol. 32, No. 4, pp. 680–689.

Schwarzkopf, M. D., and S. B. Fels, 1985: Improvements to the algorithm for computing CO2 transmissivities and cooling rates. J. Geophys. Res., 90, 541-550.

Schwarzkopf, M. D., and S. B. Fels, 1991: The simplified exchange method revisited: An accurate, rapid method for computations of infrared cooling rates and fluxes. J. Geophys. Res., 96, 9075-9096.

Skamarock, W. C., J. B. Klemp, J. Dudhia, D. O. Gill, D. M. Barker, W. Wang and J. G. Powers, 2005: A Description of the Advanced Research WRF Version 2, NCAR Tech Note, NCAR/TN–468+STR, 88 pp. [Available from UCAR Communications, P.O. Box 3000, Boulder, CO, 80307]. Available on-line at: http://box.mmm.ucar.edu/wrf/users/docs/arw_v2.pdf)

Smirnova, T. G., J. M. Brown, and S. G. Benjamin, 1997: Performance of different soil model configurations in simulating ground surface temperature and surface fluxes. Mon. Wea. Rev., 125, 1870–1884.

Smirnova, T. G., J. M. Brown, S. G. Benjamin, and D. Kim, 2000: Parameterization of cold season processes in the MAPS land-surface scheme. J. Geophys. Res., 105 (D3), 4077-4086.

Tao, W.-K., J. Simpson, and M. McCumber 1989: An ice-water saturation adjustment, Mon. Wea. Rev., 117, 231–235.

Troen, I. and L. Mahrt, 1986: A simple model of the atmospheric boundary layer: Sensitivity to surface evaporation. Boundary Layer Meteor., 37, 129-148.

Thompson, G., R. M. Rasmussen, and K. Manning, 2004: Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part I: Description and sensitivity analysis. Mon. Wea. Rev., 132, 519–542.

Wicker, L. J., and R. B. Wilhelmson, 1995: Simulation and analysis of tornado development and decay within a three-dimensional supercell thunderstorm. J. Atmos. Sci., 52, 2675–2703.